UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/923,175	08/06/2001	Bruce Alan Vessey	TN134	1709

7590 03/19/2008

MICHAEL B. Atlass, Esq.
Unisys Corporation
Unisys Way, MS/E8-114
Blue Bell, PA 19424-0001

EXAMINER	
ANYA, CHARLES E	

ART UNIT	PAPER NUMBER
2194	

MAIL DATE	DELIVERY MODE
03/19/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	09/923,175	VESSEY ET AL.	
	Examiner	Art Unit	
	Charles E. Anya	2194	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3/* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *12/07/08*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *101-138* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *101-138* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 101-138 are pending in this application.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 101-126,129,134 and 136 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 6,366,947 B1 to Kavner in view of U.S. Pat. No. 5,805,867 to Kodaira.

3. As to claim 101, Kavner teaches a method for communicating between first and second applications executing on respective first and second independent partitions of a partitionable computer system, wherein each of said first and second partitions operates under the control of a separate operating system (Web Browser 201 (Client)/Servers 121/122 Col. 12 Ln. 22 – 43), and wherein said first and second applications are configured to communicate with each other via a physical network using standard network interfaces without the need for providing an external network connection therebetween (Winsock 403 Col. 11 Ln. 61 – 67, Col. 12 Ln. 23 – 37); said method comprising: receiving a request made by said first application for establishing a network

connection with said second application for sending a message thereto; and in response to said request, establishing an emulated network connection between said first and said second applications ("…sending of calls…" Col. 11 Ln. 61 – 67, "…"get" calls…" Col. 12 Ln. 23 – 37, "…emulates connection…" Col. 13 Ln. 31 – 38, Col. 16 Ln. 20 – 36); said establishing being such that said emulation network connection permits said first and second applications to communicate with each other via said emulated network connection using standard network interfaces even though there is no physical network therebetween ("…without actually…" Col. 11 Ln. 65 – 67, "…appears that the call is going out over Internet 120…" Col. 12 Ln. 8 – 30); said establishing also being such that the receiving of a message by said second application from said first application via said emulation network connection appears to said second application as having been sent via an external physical network even though there is no external physical network ("…without actually…" Col. 11 Ln. 65 – 67, "…appears that the call is going out over Internet 120…" Col. 12 Ln. 8 – 30, "…emulates connection…" Col. 13 Ln. 31 – 38, Col. 16 Ln. 20 – 36).

Kavner is silent with reference to establishing an emulated network connection between said first and said second applications through a shared memory region of the computer system shared by said first and said second partitions.

Kodaira teaches establishing an emulated network connection between said first and said second applications through a shared memory region of the computer system shared by said first and said second partitions ("…asynchronous communication…" Col. 3 Ln. 19 – 25, Col. 12 Ln. 16 – 25, Process 41 Col. 7 Ln. 11 – 29).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the system of Kavner with teaching of Kodaira because the teaching of Kodaira would improve the system of Kavner by providing a process for exchanging data between programs running concurrently.

4. As to claim 102, Kodaira teaches the method recited in claim 101, wherein said emulated network connection requested by said first application comprises a network socket connection, and wherein the step of establishing a connection through the shared memory region comprises establishing a connection through the shared memory region that emulates a network socket connection ("…asynchronous communication…" Col. 3 Ln. 19 – 25, Col. 12 Ln. 16 – 25, Process 41 Col. 7 Ln. 11 – 29).

5. As to claim 103, Kodaira teaches the method recited in claim 101, wherein said establishing communication between said first and second applications includes: creating a data structure in said shared memory region comprising a plurality of data segments for use in sending a message from said first application to said second application (Shared Memory M1/M2 Col. 9 Ln. 18 – 29).

6. As to claim 104, Kodaira teaches the method recited in claim 103, wherein sending a message from said first application to said second application includes: writing, from the first partition on behalf of the first application, a message to one or more of said data segments, as needed ("…writing…" Col. 9 Ln. 18 – 29), and updating

an indication of the data segment containing the most recently written portion of the message ("…informs…" Col. 9 Ln. 30 – 34, Col. 10 Ln. 21 – 30, Col. 10 Ln. 49 – 54, "…flag…" Col. 10 Ln. 49 – 54, "…semaphore…" Col. 11 Ln. 6 – 18); reading from the second partition on behalf of the second application, a message from said one or more data segments ("…receives…" Col. 10 Ln. 55 – 63) and updating an indication of which data segments have been read from the data structure ("…flag is caused to be decremented…" Col. 10 Ln. 63 – 67, "…semaphore…" Col. 11 Ln. 6 – 18); and providing the message read from the data structure to the second application in accordance with an API associated with the requested network connection (Col. 10 Ln. 65 – 67).

7. As to claim 105, although the Kavner and Kodaira references do not explicitly teach the method of claim 104, wherein said plurality of data segments form a circular buffer, and wherein updating an indication of the data segment containing the most recently written portion of the message comprises incrementing a head index, the Examiner would take official notice.

8. As to claim 106, although the Kavner and Kodaira references do not explicitly teach the method of claim 105, wherein updating an indication of which data segments have been read from the data structure comprises incrementing a tail index, the Examiner would take official notice.

9. As to claim 107, Kodaira teaches the method of claim 106, further comprising polling, by the receiving partition, the shared memory region to determine if the message has been written to the shared memory region ("…flag…" Col. 10 Ln. 49 – 60, "…semaphore…" Col. 11 Ln. 6 – 18).

10. As to claim 108, Kodaira teaches the method of claim 107, further comprising receiving, by the receiving partition, an interrupt initiated by the sending partition and indicating that the message has been written to the shared memory region ("…flag…" Col. 10 Ln. 49 – 54, "…semaphore…" Col. 11 Ln. 6 – 18).

11. As to claim 109, Kavner teaches the method of claim 102, wherein the step of establishing an emulated socket connection between the first partition and the second partition further comprises performing the following steps on the second partition: (a) creating an emulated socket connection on behalf of the second application to listen for attempts to connect thereto (Steps 520/522 Col. 13 Ln. 26 – 30); **while Kodaira teaches** (b) receiving a connect message from the first partition that identifies a memory location of the shared memory region at which the first partition has allocated a first data area to serve as a buffer for transferring data from the first partition to the second partition ("…queue ID…" Col. 10 Ln. 42 – 67); (c) matching the received connect message to the listening socket created in step (a) ("…one of them matches…" Col. 11 Ln. 58 – 62); (d) allocating a second data area in the shared memory region to serve as a buffer for transferring data from the second partition to the first partition

(Process 41 Col. 9 Ln. 36 – 38, Execution Address Trigger Setting Portion 74 Col. 11 Ln. 34 – 41); (e) mapping both the first and second data areas into a process space of the listening socket ("…mapped…" Col. 9 Ln. 18 – 22); (f) initializing the second data area (Process 41 Col. 9 Ln. 36 – 38); and (g) returning a connected indication to the first partition ("…transmits condition…" Col. 9 Ln. 35 – 38) and informing the application on the second partition that the emulated socket connection has been established (the Examiner would take official notice).

12. As to claim 110, Kavner teaches the method of claim 109, further comprising performing the following steps on the first partition: (a') receiving the request from the first application to establish the socket connection with the second application (Step 520 Col. 13 Ln. 36 – 30); (b') creating an emulated connecting socket (Step 522 Col. 13 Ln. 26 – 30); **while Kodaira teaches** (c') allocating the first data area in the shared memory region (Process 41 Col. 9 Ln. 36 – 38, Execution Address Trigger Setting Portion 74 Col. 11 Ln. 34 – 41); (d') sending the connect message to the second partition that identifies the memory location of the shared memory region at which the first data area has been allocated ("…queue ID…" Col. 10 Ln. 42 – 67); and (e') upon receipt of the connected indication from the second partition, mapping the first and second data areas into a process space of the connecting emulated socket to establish the emulated socket connection between the first and second partitions ("…flag…" Col. 10 Ln. 49 – 54, "…semaphore…" Col. 11 Ln. 6 – 18).

13. As to claim 111, Kodaira teaches the method of claim 101, further comprising: creating in the shared memory region, a plurality of output queues, one for each of said first second partitions, the output queue for a given partition indicating whether that partition has placed in the shared memory region a message intended for any of the other partitions and if so, identifying a buffer containing the message ("…queue ID…" Col. 10 Ln. 42 – 67, Process 41 Col. 9 Ln. 36 – 38, Execution Address Trigger Setting Portion 74 Col. 11 Ln. 34 – 41), each partition polling the output queues of other partitions to determine whether those other partitions have placed any messages intended for it in the shared memory region ("…checks the signal flag…" Col. 10 Ln. 55 – 60); receiving at the first partition from the first application, said request to send a message to the second application via the emulated network connection ("…receives…" Col. 10 Ln. 55 – 67); writing, in response to the received request, the message to an available buffer in the shared memory region and indicating in the output queue of the first partition that the message has been written thereto ("…transmit its condition…" Col. 34 – 38); determining, at the second partition, from the output queue of the first partition, that the message has been placed in the available buffer and retrieving the message from the available buffer; and providing the message read from the data structure to the second application in accordance with an API associated with the requested network connection ("…checks the signal flag…" Col. 10 Ln. 55 – 60, Col. 11 Ln. 57 – 64, Col. 12 Ln. 26 – 37).

14. As to claim 112-122, see the rejection of claims 101-111 respectively.

15. As to claim 123, Kavner teaches a computer system comprising: a plurality of processing modules, groups of one or more processing modules being configured so as to provide at least first and second separate independent partitions within the computer system, each of said first and second partitions operating under the control of a separate operating system, said first partition providing for executing a first application and said second partition providing for executing a second application (Web Browser 201 (Client)/Servers 121/122 Col. 12 Ln. 22 – 43, each of said first and second applications being configured to communicate with the other application via a physical network using standard network interfaces (Winsock 403 Col. 11 Ln. 61 – 67, Col. 12 Ln. 23 – 37, "…without actually…" Col. 11 Ln. 65 – 67, "…appears that the call is going out over Internet 120…" Col. 12 Ln. 8 – 30), such that said first and second applications communicate with each other using standard network interfaces even though there is no physical network there between ("…without actually…" Col. 11 Ln. 65 – 67, "…appears that the call is going out over Internet 120…" Col. 12 Ln. 8 – 30), and also such that the receiving of a message by said second application from said first application appears to said second application as having been sent via an external physical network even though there is no external physical network ("…without actually…" Col. 11 Ln. 65 – 67, "…appears that the call is going out over Internet 120…" Col. 12 Ln. 8 – 30).

Kavner is silent with reference to a main memory to which each processing module is connected, the main memory having defined therein at least one shared memory region to which at least said first and second ones of said partitions have shared access; and program code, executing on each of at least said first partition and

said second partition of the computer system, said program code establishing a connection between a said first application on said first partition and said second application on said second partition through the shared memory region, wherein the connection through the shared memory region emulates a network connection requested by one of said applications; wherein said program code executing on each of said first and second partitions comprises a shared memory service provider that serves as an interface between a component of the computer system that provides an API through which said first application can make said request for a network connection and the shared memory region of the main memory through which the emulated network connection is established; wherein the shared memory service provider on each of said first and second partitions establishes a data structure in the shared memory region through which data is transferred from that partition to the shared memory service provider on the other partitions; wherein the data structure comprises: a plurality of data segments, each of the plurality of data segments for storing network message data to be sent from a sending shared memory service provider to a receiving shared memory service provider; a control segment for controlling reading and writing of data in the plurality of data segments, the control segment comprising: a first portion comprising: a first field for storing an indication of the data segment containing the most recently written network message data; and a second field for storing an indication of the data segment containing the earliest written, but to read, network message data; and a plurality of second portions, each second portion corresponding to one of the plurality of data segments for control of the data segment, each second portion comprising: a first

field for storing an indication of the beginning of network message data within the data segment; and a second field for storing an indication of the end of network message data within the data segment.

Kodaira teaches a main memory to which each processing module is connected, the main memory having defined therein at least one shared memory region to which at least said first and second ones of said partitions have shared access ("…asynchronous communication…" Col. 3 Ln. 19 – 25, Col. 12 Ln. 16 – 25, Process 41 Col. 7 Ln. 11 – 29); and program code, executing on each of at least said first partition and said second partition of the computer system, said program code establishing a connection between a said first application on said first partition and said second application on said second partition through the shared memory region, wherein the connection through the shared memory region emulates a network connection requested by one of said applications ("…asynchronous communication…" Col. 3 Ln. 19 – 25, Col. 12 Ln. 16 – 25, Process 41 Col. 7 Ln. 11 – 29); wherein said program code executing on each of said first and second partitions comprises a shared memory service provider that serves as an interface between a component of the computer system that provides an API through which said first application can make said request for a network connection and the shared memory region of the main memory through which the emulated network connection is established (Shared Memories Supervisory Portion 46/Process 41 Col. 7 Ln. 30 – 43, Col. 9 Ln. 18 – 38); wherein the shared memory service provider on each of said first and second partitions establishes a data structure in the shared memory region through which data is transferred from that partition to the shared memory

service provider on the other partitions (Shared Memories Supervisory Portion 46/Process 41 Col. 9 Ln. 18 – 38); wherein the data structure comprises: a plurality of data segments, each of the plurality of data segments for storing network message data to be sent from a sending shared memory service provider to a receiving shared memory service provider (Shared Memories M1/M2 Col. 9 Ln. 18 – 29); a control segment for controlling reading and writing of data in the plurality of data segments, the control segment comprising: a first portion comprising: a first field for storing an indication of the data segment containing the most recently written network message data ("…queue ID…" Col. 10 Ln. 21 – 30); and a second field for storing an indication of the data segment containing the earliest written, but not read, network message data (the Examiner would take official notice); and a plurality of second portions, each second portion corresponding to one of the plurality of data segments for control of the data segment, each second portion comprising: a first field for storing an indication of the beginning of network message data within the data segment; and a second field for storing an indication of the end of network message data within the data segment (the Examiner would take official notice).

16. As to claim 124, Kodaira teaches the computer system recited in claim 123, wherein the first portion further comprises: a third field for storing an indication that the sending shared memory service provider is waiting to send the network message ("…flag…" Col. 10 Ln. 49 – 54, "…semaphore…" Col. 11 Ln. 6 – 18); and a fourth field for storing an indication that the receiving shared memory service provider is waiting to

receive the network message ("…flag…" Col. 10 Ln. 49 – 54, "…semaphore…" Col. 11 Ln. 6 – 18).

17. As to claim 125, although the Kavner and Kodaira references do not explicitly teach the computer system recited in claim 123, wherein each second portion further comprises a third field for storing an indication of a length of network message data within the data segment, the Examiner would take official notice.

18. As to claim 126, although the Kavner and Kodaira references do not explicitly teach the computer system recited in claim 125, wherein the plurality of data segments are linked to form a circular buffer, and wherein each second portion further comprises: a fourth field for storing an indication of the next data segment in the circular buffer, and a fifth field for storing an indication that the data segments contains a last portion of a network message stored across a plurality of data segments, the Examiner would take official notice.

19. As to claim 129, Kodaira teaches the computer system recited in claim 123, wherein the connection established through the shared memory region emulates a network socket connection ("…asynchronous communication…" Col. 3 Ln. 19 – 25, Col. 12 Ln. 16 – 25, Process 41 Col. 7 Ln. 11 – 29).

20. As to claims 134 and 136, see the rejection of claims 107 and 111 respectively.

21. Claims 127,128,130-133,135,137 and 138 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 6,366,947 B1 to Kavner in view of U.S. Pat. No. 5,805,867 to Kodaira as applied to claim 123 above, and further in view of U.S. Pat. No. 6,658,469 B1 to Massa et al.

22. As to claim 127, Kodaira and Kavner are silent with reference to the computer system recited in claim 123, wherein the computer system provides a resource through which the shared memory service provider can establish the data structure and control the transfer of data through it, the resource providing the ability to perform at least one of the following operations on the shared memory region: (i) allocate an area of the shared memory region; (ii) map and unmap an area of the shared memory region; deallocate an area of the shared memory region; (iii) send and receive signals to and from other partitions via the shared memory region; and (iv) receive status information about the shared memory region and about selected partitions.

Massa teaches the computer system recited in claim 123, wherein the computer system provides a resource through which the shared memory service provider can establish the data structure ("…memory buffer…" Col. 8 Ln. 1 – 13) and control the transfer of data through it, the resource providing the ability to perform at least one of the following operations on the shared memory region: (i) allocate an area of the shared memory region ("…register a memory buffer…" Col. 8 Ln. 1 – 13); (ii) map and unmap an area of the shared memory region ("…register a memory buffer…deregister a

memory buffer…" Col. 8 Ln. 1 – 13); deallocate an area of the shared memory region ("…deregister a memory buffer…" Col. 8 Ln. 1 – 13); (iii) send and receive signals to and from other partitions via the shared memory region ("…transfer data…" Col. 8 Ln. 1 – 13); and **while Kodaira teaches** (iv) receive status information about the shared memory region and about selected partitions ("…flag…" Col. 10 Ln. 49 – 54, "…semaphore…" Col. 11 Ln. 6 – 18).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the system of Kodaira and Kavner with teaching of Massa because the teaching of Massa would improve the system of Kodaira and Kavner by establishing communication through an alternative transport provider and therefore emulating the semantics of the primary transport provider such that communicating applications are unaware that the alternative transport provider is in use (Massa Col. 3 Ln. 48 – 52).

23. As to claim 128, Massa teaches the computer system recited in claim 123, wherein the shared memory service provider comprises: a dynamic link library (DLL) that executes in a user mode of the operating system of its respective partition, there being an instance of the shared memory service provider DLL in a process space of each application in the partition that may request the establishment of a network connection ("…dynamic link library (dll) Col. 7 Ln. 60 – 67); and a device driver that executes in a kernel mode of the operating system of the respective partition, there

being only one instance of the device driver in each partition (TCP/IP Driver 110 Col. 10 Ln. 46 – 51).

24. As to claim 130, Massa teaches the computer system recited in claim 128, wherein said program code executing on each of said at least first and second partitions comprises a shared memory service provider that serves as an interface between a component of the computer system that provides an API through which an application can make a request for a network socket connection and the shared memory region of the main memory through which the emulated network socket connection is established (Col. 7 Ln. 60 – 67, "…Winsock service provider…" Col. 8 Ln. 1 – 13).

25. As to claim 131, Massa teaches the computer system recited in claim 129, wherein the operating system in each partition comprises a MICROSOFT WINDOWS operating system ("…Windows NT operating system…" Col. 6 Ln. 66 – 67, Col. 7 Ln. 1 – 4), and wherein the component of the computer system that provides the API of the requested socket Connection comprises a Winsock DLL and a Winsock Switch (Switch 86…dynamic link library (dll)…" Col. 7 Ln. 60 – 67), the Winsock DLL forwarding a request for a socket connection made by an application in a given partition to the Winsock Switch, which Winsock Switch allows multiple service providers (SAN Provider 106/TCP/IP Provider 110 Col. 11 Ln. 1 – 9), each of which provide TCP/IP services, to service such a request, and wherein the shared memory service provider acts as a TCP/IP service provider so that a request from an application for a socket connection

can be serviced by the shared memory service provider (Switch 86 Col. 8 Ln. 1 – 13, Local Switch 120/Remote Switch 126 Col. 11 Ln. 10 – 30).

26. As to claims 132,133 and 135, see the rejection of claims 109,107 and 111 respectively.

27. As to claim 137, Massa teaches the computer system recited in claim 135, wherein the program code executing on each of said first and second partitions further comprises a shared memory driver that receives a request to send a message to an application on another partition, the request having been made in accordance with the application programming interface (API) associated with the requested type of network connection, and that, in response to the request, causes the message to be placed in an available buffer in the shared memory region ("…TCP/IP driver…" Col. 10 Ln. 44 – 48, Col. 11 Ln. 10 – 20) and **while Kodaira teaches** causes an indication of the message to be placed in the output queue of the sending partition ("…flag…" Col. 10 Ln. 49 – 54, "…semaphore…" Col. 11 Ln. 6 – 18).

28. As to claim 138, Massa teaches the computer system recited in claim 136, wherein the shared memory driver on each partition implements a same interface as a network device driver to enable application programs and the operating system on that partition to send communications to other partitions via the shared memory region in the same manner that communications are sent to other computer systems over a network

via a network interface card (Col. 9 Ln. 35 – 50, "…TCP/IP driver…" Col. 10 Ln. 44 – 48, Col. 11 Ln. 10 – 20).

Response to Arguments

Applicant's arguments filed 12/7/07 have been fully considered but they are not persuasive.

Applicant argues in substance that (1) the Kavner prior art disclosure includes an external network and as such teaches away from the instant invention (2) the emulation network connection of the Kavner prior art is not for permitting communication between network configured applications running on different partitions so that they can communicate with each other even though there is not external network present and (3) the Kodaira prior art does not disclose an emulation network connection.

The Examiner respectfully traverses Applicant arguments:

As to point (1), the fact that the Kavner prior art discloses an external network does not negate the fact that emulated network connection is also taught. For instance the instance application in claim 101 includes a first and second applications configured to communicate with each other using a **physical network and emulated network connection.** If the instance application includes physical network and emulated network connection why would it be out of place for the Kavner prior art to teach the same.

As to point (2), contrary to Applicant's assertion the Kavner prior art does teach emulation network connection for permitting communication between networks configured applications running on different partitions so that they can communicate

with each other. Winsock which provides the emulation network connection is a technical specification that defines how windows networks software should access network services, especially TCP/IP. The network services referred to here are provided by applications running in a server computer to applications running in a client computer (network software)/(step 532 Col. 13 Ln. 31 – 38), hence it is safe to say that the Kavner prior art teaches emulation network connection (via the Winsock) for permitting communication between networks configured applications running on different partitions so that they can communicate with each other.

As to point (3), applicant's argument moot since the Kavner prior art discloses emulation network connection.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

U.S. Pat. No. 5,870,589 to Alexander, Jr. et al.: directed to a method for enhancing performance in communications involving one or more emulated networks.

U.S. Pat. No. 5,894,566 to Croslin: directed to a system and method for emulating network outages.

U.S. Pat. No. 5,935,212 to Kalajan et al.: directed to connection oriented session emulation.

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Charles E. Anya whose telephone number is 571-272-3757. The examiner can normally be reached on 8:30-5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Meng-Ai An/
Supervisory Patent Examiner, Art Unit 2195

cea.